Algorithms for stochastic nonconvex and nonsmooth optimization

Courtney Paquette

University of Waterloo

Mathematics and Statistics, McGill University
February 4, 2019
Research Directions

(1). Thesis Work

- computational complexity
- nonsmooth analysis of eigenvalues
- composite nonlinear models \((h \circ c)\)
- statistical guarantees for nonconvex problems

(2). Post doc

- stochastic optimization
- constrained conjugate gradient
Research Directions

(1). Thesis Work
   ▶ computational complexity
   ▶ nonsmooth analysis of eigenvalues
   ▶ composite nonlinear models \((h \circ c)\)
   ▶ statistical guarantees for nonconvex problems

(2). Post doc
   ▶ stochastic optimization
   ▶ constrained conjugate gradient

(1). Local search for non-smooth and non-convex problems
(2). Adaptive line search for stochastic optimization
Local search for non-smooth and non-convex problems

Joint work with D. Davis (Cornell), D. Drusvyatskiy (U. Washington), and K. MacPhee (U. Washington)
Why study nonsmooth and nonconvex optimization?

\[
\min_x g(x)
\]

Nonsmooth and nonconvex losses arise often...

- Structure (sparsity), robustness (outliers), stability (better conditioning)

Common problem class: \((\text{convex}) \circ (\text{smooth})\)

(Fletcher ’80, Powell ’83, Burke ’85, Wright ’90, Lewis-Wright ’08, Cartis-Gould-Toint ’11)
Example: Robust Phase Retrieval

**Problem:** Find $x \in \mathbb{R}^d$ such that

$$(a_i^T x)^2 \approx b_i \quad a_1, \ldots, a_m \in \mathbb{R}^d, \quad b_1, \ldots, b_m \in \mathbb{R}.$$ 

**Composite formulation:**

$$\min_x g(x) := \frac{1}{m} \sum_{i=1}^{m} |(a_i^T x)^2 - b_i|$$

**Assumptions:**

$m \gtrsim d, \quad b_i = (a_i^T \bar{x})^2$ for some $\bar{x} \in \mathbb{R}^d, \quad a_i \sim N(0, I_d)$ and independent
Example: Robust Phase Retrieval

Problem: Find $x \in \mathbb{R}^d$ such that

$$(a_i^T x)^2 \approx b_i \quad a_1, \ldots, a_m \in \mathbb{R}^d, \quad b_1, \ldots, b_m \in \mathbb{R}.$$ 

Composite formulation:

$$\min_x g(x) := \frac{1}{m} \sum_{i=1}^{m} |(a_i^T x)^2 - b_i|$$

Assumptions:

$m \gtrsim d$, $b_i = (a_i^T \bar{x})^2$ for some $\bar{x} \in \mathbb{R}^d$, $a_i \sim N(0, I_d)$ and independent

Consequences: \( \exists \) constants $\rho, \tau > 0$ such that with probability $1 - e^{-cm}$

- **Weakly-convex:** (Duchi-Ruan ’17)
  $$x \mapsto g(x) + \frac{\rho}{2} \|x\|_2^2$$ is convex

- **Sharpness:** (Eldar-Mendelson ’14)
  $$g(x) \geq \tau \|\bar{x}\|_2 \text{dist}(x, \{\pm \bar{x}\}).$$

  Holds even when up to 1/2 the points are corrupted!
Intuition

\[ g(x) = \frac{1}{m} \sum_{i=1}^{m} |(a_i^T x)^2 - b_i| \] approximates the population objective:

\[ g_P(x) = \mathbb{E}_{a \sim \mathcal{N}}[(a^T x)^2 - (a^T \bar{x})^2] \]
Local search

\[
\min_x g(x)
\]

**Strategy:**
- Find a moderately accurate solution \( \hat{x} \) at a low sample complexity cost
  - Available for: phase retrieval (Candès et al. ’15), blind deconvolution (Ma et al. ’17, Li et al. ’18), matrix sensing (Boczar et al. ’16)
- Refine \( \hat{x} \) with a rapidly converging algorithm
Local search

\[
\min_x g(x)
\]

\(g\) is \(\mu\)-sharp and \(\rho\)-weakly convex

**Strategy:**

- Find a moderately accurate solution \(\hat{x}\) at a low sample complexity cost
  - Available for: phase retrieval (Candès et al. ’15), blind deconvolution (Ma et al. ’17, Li et al. ’18), matrix sensing (Boczar et al. ’16)
- Refine \(\hat{x}\) with a rapidly converging algorithm

Is there a generic gradient-based **local search procedure** for nonsmooth and nonconvex problems?
Stationary points

- **Subdifferential** \( \partial g(x) \) consists of \( v \in \mathbb{R}^d \) such that

\[
g(y) \geq g(x) + \langle v, y - x \rangle + o(||y - x||), \quad \text{as } y \to x
\]

- Under weak-convexity,

\[
x \text{ stationary} \iff \inf_{||v||=1} g'(x, v) \geq 0 \iff 0 \in \partial g(x).
\]

**Remark:**

\[
\partial(h \circ c)(x) = \nabla c(x)^T \partial h(c(x))
\]
Stationary points

- **Subdifferential** $\partial g(x)$ consists of $v \in \mathbb{R}^d$ such that
  \[ g(y) \geq g(x) + \langle v, y - x \rangle + o(\|y - x\|), \quad \text{as } y \to x \]

- Under weak-convexity,
  \[ x \text{ stationary } \iff \inf_{\|v\|=1} g'(x, v) \geq 0 \iff 0 \in \partial g(x). \]

**Remark:**

\[
\partial (h \circ c)(x) = \nabla c(x)^T \partial h(c(x))
\]

**Lemma (Davis-Drusvyatskiy-MacPhee-P)**

No extraneous stationary points of $g$ lie in the tube:

\[
\mathcal{T} := \left\{ x \in \mathbb{R}^d : \text{dist}(x; \mathcal{S}) < \frac{\mu}{\rho} \right\}
\]
Meta-Theorem:
Simple algorithms for **sharp** and **weakly convex** functions converge rapidly
Meta-Theorem:
Simple algorithms for **sharp** and **weakly convex** functions converge rapidly

\[
\min_x g(x) := h(c(x)), \quad \text{(convex)} \circ \text{(smooth)}
\]

**Prox-linear method:**

\[
x^+ := \arg\min_y \left\{ h\left(c(x) + \nabla c(x)(y - x)\right) + \frac{\rho}{2} \|y - x\|^2 \right\}
\]

(Burke ’85, ’91, Fletcher ’82, Powell ’84, Wright ’90, Yuan ’83, Cartis-Gould-Toint ’11)

Eg: proximal gradient, Levenberg-Marquardt

**Subgradient method:**

\[
x^+ = x - \left(\frac{g(x) - \inf_v g}{\|v\|^2}\right) v \quad \text{where } v \in \partial g(x)
\]
Prox-linear method

\[ x^+ = \arg\min_y \left\{ h(c(x) + \nabla c(x)(y - x)) + \frac{\rho}{2} \|y - x\|^2 \right\} \]

**Thm:** (Burke-Ferris ’93)

Suppose that \( g = h \circ c \) is
- \( \rho \)-weakly convex
- \( \mu \)-sharp
- \( \text{dist}(x_0, S) \leq \frac{\mu}{2\rho} \)

Then

\[ \text{dist}(x_{k+1}, S) \leq \frac{\mu}{\rho} \cdot \left(\frac{1}{2}\right)^{2^k} \quad \text{for all } k. \]

Global convergence guarantees (Drusvyatskiy-P, Math. Program ’16)

**Eg:** phase retrieval
- \( \frac{\mu}{\rho} \) is **dimension independent** w.h.p. (Eldar-Mendelson ’14, Duchi-Ruan ’17)
Subgradient Methods

### Polyak subgradient method:

\[ x^+ = x - \left( \frac{g(x) - \inf_{v} g}{\|v\|^2} \right) v \quad \text{where } v \in \partial g(x). \]

**Thm:** (Polyak ’67, Davis-Drusvyatskiy-MacPhee-P ’17)

Suppose that \( g \) is

- \( \rho \)-weakly convex
- \( L \)-Lipschitz
- \( \mu \)-sharp
- \( \text{dist}(x_0, S) \leq \frac{\mu}{2\rho} \)

Then

\[
\frac{\text{dist}(x_{k+1}, S)}{\text{dist}(x_k, S)} \leq \sqrt{1 - \left( \frac{\mu}{L \sqrt{2}} \right)^2}, \quad \text{for all } k.
\]

**Eg:** phase retrieval

- \( \frac{\mu}{\rho} \), \( \frac{\mu}{L} \) are dimension independent w.h.p. (Eldar-Mendelson ’14, Davis-Drusvyatskiy-MacPhee-P ’17)
Figure: $(d, m) \approx (2^{23}, 2^{24})$. Iteration 1.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 2.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 3.
Figure: $(d, m) \approx (2^{23}, 2^{24})$. Iteration 4.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 5.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 6.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 7.
Figure: $(d, m) \approx (2^{23}, 2^{24})$. Iteration 8.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 9.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 10.
Figure: $(d, m) \approx (2^{23}, 2^{24})$. Iteration 11.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 12.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 13.
Figure: \((d, m) \approx \left(2^{23}, 2^{24}\right)\). Iteration 14.
Figure: \((d, m) \approx (2^{23}, 2^{24})\). Iteration 15.
Figure: Convergence plot (iterates vs. $\|x_k - \bar{x}\|/\|\bar{x}\|$).
Research Program:

Physical/Statistical well-posedness $\Rightarrow$ (convex) $\circ$ (smooth) with “sharpness”-like conditions $\Rightarrow$ Simple algorithms converge rapidly

- Robust phase retrieval (Duchi-Ruan ’17, Davis-Drusvyatskiy-MacPhee-P ’17)
- Blind deconvolution/bi-convex sensing (Ling-Strohmer ’15, Ahmed et al. ’14)
- Covariance estimation (Chen et. al ’15)
- Robust PCA (Candes et al. ’11, Chandrasekaran et al. ’11, Netrapalli et al. ’14)
- Non-negative matrix factorization (Lee-Seung ’99, Donoho-Stodden ’03)
Adaptive line search method for smooth stochastic optimization

Joint work with K. Scheinberg
Stochastic optimization

Central task in machine learning

$$\min_x f(x) = \mathbb{E}_{\xi \sim P}[\tilde{f}(x; \xi)]$$

Stochastic gradient descent (SGD):

$$\begin{cases}
\text{Pick } \xi \sim P \\
\text{Set } x_{k+1} = x_k - \alpha_k \nabla \tilde{f}(x_k; \xi)
\end{cases}$$

- **Major drawback:** stepsize, $\alpha_k$, requires lots of tuning
Stochastic optimization

Central task in machine learning

\[
\min_x f(x) = \mathbb{E}_{\xi \sim P}[\tilde{f}(x; \xi)]
\]

Stochastic gradient descent (SGD):

\[
\begin{cases}
\text{Pick } \xi \sim P \\
\text{Set } x_{k+1} = x_k - \alpha_k \nabla \tilde{f}(x_k; \xi)
\end{cases}
\]

- **Major drawback:** stepsize, \(\alpha_k\), requires lots of tuning

Deterministic setting: Use line search techniques

**Question:**
Can the line search technique be adapted to the stochastic setting?
(Deterministic) Backtracking Line Search

Classical problem

\[
\min_{x \in \Omega} f(x)
\]

\(f : \Omega \rightarrow \mathbb{R}\) with \(L\)-Lipschitz gradient

**Gradient descent:** \(x_{k+1} = x_k - \alpha \nabla f(x_k), \quad \alpha \in (0, 1/L]\)
(Deterministic) Backtracking Line Search

Classical problem

$$\min_{x \in \Omega} f(x)$$

$$f : \Omega \rightarrow \mathbb{R}$$ with $L$-Lipschitz gradient

**Gradient descent:** $x_{k+1} = x_k - \alpha \nabla f(x_k), \quad \alpha \in (0, 1/L]$

**Backtracking Line Search Algorithm**

- Compute $f(x_k)$ and $\nabla f(x_k)$
- Check sufficient decrease *(Armijo ’66)*

$$f(x_k - \alpha_k \nabla f(x_k)) \leq f(x_k) - \theta \alpha_k \| \nabla f(x_k) \|^2$$

- Successful: $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$ and **increase** $\alpha_k \Rightarrow \alpha_{k+1} = \gamma^{-1} \alpha_k$
- Unsuccessful: $x_{k+1} = x_k$ and **decrease** $\alpha_k \Rightarrow \alpha_{k+1} = \gamma \alpha_k$
(Deterministic) Backtracking Line Search

Classical problem

\[
\min_{x \in \Omega} f(x)
\]

\(f : \Omega \to \mathbb{R}\) with \(L\)-Lipschitz gradient

Gradient descent: \(x_{k+1} = x_k - \alpha \nabla f(x_k), \quad \alpha \in (0, 1/L]\)

Backtracking Line Search Algorithm

- Compute \(f(x_k)\) and \(\nabla f(x_k)\)
- Check sufficient decrease (Armijo ’66)

\[
\underbrace{f(x_k - \alpha_k \nabla f(x_k))}_{\text{function value at next step}} \leq \underbrace{f(x_k) - \theta \alpha_k \|\nabla f(x_k)\|^2}_{\text{linearization of } f \text{ at current step}}
\]

- Successful: \(x_{k+1} = x_k - \alpha_k \nabla f(x_k)\) and increase \(\alpha_k \Rightarrow \alpha_{k+1} = \gamma^{-1} \alpha_k\)
- Unsuccessful: \(x_{k+1} = x_k\) and decrease \(\alpha_k \Rightarrow \alpha_{k+1} = \gamma \alpha_k\)
Question
Can the line search technique be adapted to stochastic setting using only knowable quantities?

Knowable quantities: e.g. bound on variance of $\nabla \tilde{f}$, $\tilde{f}$
Related works

**Line search & heuristics** Previous work requires: $\nabla f(x), \alpha_k \to 0$

- Mahsereci and Hennig; “Probabilistic line search for stochastic optimization” (JMLR 2018; NIPS 2015)
Stochastic backtracking line search

- Compute **stochastic** estimates $\nabla f(x_k)$, $f(x_k)$, and $f(x_k - \alpha_k g_k)$
- Check sufficient decrease (Armijo ’66)

$$f^+_k \leq f_k - \theta \alpha_k \|g_k\|^2$$
- Successful: $x_{k+1} = x_k - \alpha_k g_k$ and **increase** $\alpha_k \Rightarrow \alpha_{k+1} = \gamma^{-1} \alpha_k$
- Unsuccessful: $x_{k+1} = x_k$ and **decrease** $\alpha_k \Rightarrow \alpha_{k+1} = \gamma \alpha_k$
Stochastic backtracking line search

- Compute stochastic estimates $g_k, f_k, \text{ and } f^+_k$
  \[ \nabla f(x_k), f(x_k), f(x_k - \alpha_k g_k) \]

- Check sufficient decrease (Armijo ’66)
  \[ f^+_k \leq f_k - \theta \alpha_k \|g_k\|^2 \]

- Successful: $x_{k+1} = x_k - \alpha_k g_k$ and increase $\alpha_k \Rightarrow \alpha_{k+1} = \gamma^{-1} \alpha_k$

- Unsuccessful: $x_{k+1} = x_k$ and decrease $\alpha_k \Rightarrow \alpha_{k+1} = \gamma \alpha_k$

Challenges

\[ f^+_k \leq f_k - \theta \alpha_k \|g_k\|^2 \quad \Rightarrow \quad f(x_k - \alpha_k g_k) \leq f(x_k) - \theta \alpha_k \|\nabla f(x_k)\|^2 \]

- Bad function estimates may ↑ objective value

  Increase at most $\alpha_k^2 \|g_k\|^2$

- Stepsizes, $\alpha_k$, become arbitrarily small
Stochastic line search

Algorithm

- Compute random estimate of the gradient, $g_k$
- Compute random estimate of $f_k \approx f(x_k)$ and $f_k^+ \approx f(x_k - \alpha_k g_k)$
- Check the stochastic sufficient decrease

$$f_k^+ \leq f_k - \theta \alpha_k \|g_k\|^2$$

- Successful: $x_{k+1} = x_k - \alpha_k g_k$ and $\alpha_k \uparrow \Rightarrow \alpha_{k+1} = \gamma^{-1} \alpha_k$

  - Reliable step: If $\alpha_k \|g_k\|^2 \geq \delta_k^2$, $\uparrow \delta_k \Rightarrow \delta_{k+1} = \gamma^{-1} \delta_k^2$
  - Unreliable step: If $\alpha_k \|g_k\|^2 < \delta_k^2$, $\downarrow \delta_k \Rightarrow \delta_{k+1} = \gamma \delta_k^2$

- Unsuccessful: $x_{k+1} = x_k$, decrease $\alpha_k$, and decrease $\delta_k$

  $\Rightarrow \alpha_{k+1} = \gamma \alpha_k$ and $\delta_{k+1}^2 = \gamma \delta_k^2$. 


Randomness assumptions

- **Accurate gradient** $g_k$ w/ prob. $p_g$:

  $$\Pr\left(\|g_k - \nabla f(x_k)\| \leq \alpha_k \|g_k\|\right) \geq p_g$$

- **Accurate function estimates** $f_k$ and $f_k^+$ w/ prob. $p_f$:

  $$\Pr\left(|f(x_k) - f_k| \leq \alpha_k^2 \|g_k\|^2\right)$$
  and
  $$\Pr\left(|f(x_k - \alpha_k g_k) - f_k^+| \leq \alpha_k^2 \|g_k\|^2\right) \geq p_f$$
Randomness assumptions

- **Accurate gradient** $g_k$ w/ prob. $p_g$:

  $$\Pr(\|g_k - \nabla f(x_k)\| \leq \alpha_k \|g_k\|) \geq p_g$$

- **Accurate function estimates** $f_k$ and $f_k^+$ w/ prob. $p_f$:

  $$\Pr(|f(x_k) - f_k| \leq \alpha_k^2 \|g_k\|^2)$$
  and $$|f(x_k - \alpha_k g_k) - f_k^+| \leq \alpha_k^2 \|g_k\|^2 \geq p_f$$

- **Variance condition**

  $$\mathbb{E}[|f_k - f(x_k)|^2] \leq \theta^2 \delta_k^4$$
  (same for $f_k^+$).

**Question:** How to choose these probabilities $(p_f, p_g)$ large enough?

$p_f, p_g \geq 1/2$ at least, but $p_f$ should be large.
Satisfying randomness assumptions

\[ \min_x f(x) = E_{\xi \sim P}[\tilde{f}(x; \xi)] \]

and bound on variance

\[ E_{\xi \sim P}(\|\nabla \tilde{f}(x, \xi) - \nabla f(x)\|^2) \leq V_g, \quad E_{\xi \sim P}(|\tilde{f}(x; \xi) - f(x)|^2) \leq V_f. \]
Satisfying randomness assumptions

$$\min_x f(x) = \mathbb{E}_{\xi \sim P} [\tilde{f}(x; \xi)]$$

and bound on variance

$$\mathbb{E}_{\xi \sim P} (\| \nabla \tilde{f}(x, \xi) - \nabla f(x) \|^2) \leq V_g, \quad \mathbb{E}_{\xi \sim P} (| \tilde{f}(x; \xi) - f(x) |^2) \leq V_f.$$

Example: sampling

$$g_k = \frac{1}{|\mathcal{S}_g|} \sum_{i \in \mathcal{S}_g} \nabla \tilde{f}(x_k; \xi_i), \quad f_k = \frac{1}{|\mathcal{S}_f|} \sum_{i \in \mathcal{S}_f} \tilde{f}(x_k; \xi_i).$$

How many samples do we need?
Satisfying randomness assumptions

$$\min_x f(x) = \mathbb{E}_{\xi \sim P}[\tilde{f}(x; \xi)]$$

and bound on variance

$$\mathbb{E}_{\xi \sim P}(\|\nabla \tilde{f}(x, \xi) - \nabla f(x)\|^2) \leq V_g, \quad \mathbb{E}_{\xi \sim P}(|\tilde{f}(x; \xi) - f(x)|^2) \leq V_f.$$ 

Example: sampling

$$g_k = \frac{1}{|S_g|} \sum_{i \in S_g} \nabla \tilde{f}(x_k; \xi_i), \quad f_k = \frac{1}{|S_f|} \sum_{i \in S_f} \tilde{f}(x_k; \xi_i).$$

How many samples do we need?

Chebyshev Inequality

$$|S_g| \approx \tilde{O}\left(\frac{V_g}{\alpha_k^2 \|g_k\|^2}\right), \quad |S_f| \approx \tilde{O}\left(\max\left\{\frac{V_f}{\alpha_k^4 \|g_k\|^4}, \frac{V_f}{\delta_k^4}\right\}\right)$$
Convergence result

Key observations

- \( \Phi_k = \nu(f(x_k) - \inf f) + (1 - \nu)\alpha_k \| \nabla f(x_k) \|^2 + (1 - \nu)\theta\delta_k^2 \)
  balance each other

- \( T_\varepsilon = \inf\{ k \geq 0 : \| \nabla f(x_k) \| < \varepsilon \} \)
Convergence result

**Key observations**

- $\Phi_k = \nu(f(x_k) - \inf f) + (1 - \nu)\alpha_k \|\nabla f(x_k)\|^2 + (1 - \nu)\theta\delta_k^2$
  - balance each other

- $T_\epsilon = \inf\{k \geq 0 : \|\nabla f(x_k)\| < \epsilon\}$

**Thm: (P-Scheinberg ’18)** If $p_g p_f > 1/2$ and $p_f$ sufficiently large, then

$$
\mathbb{E}[\Phi_{k+1} - \Phi_k | \text{past}] \leq -\left(\alpha_k \|\nabla f(x_k)\|^2 + \theta\delta_k^2\right)
$$

and consequently,

$$
\mathbb{E}[T_\epsilon] \leq O\left(\frac{1}{\epsilon^2}\right).
$$
Convergence result

Key observations

- $\Phi_k = \nu (f(x_k) - \inf f) + (1 - \nu) \alpha_k \| \nabla f(x_k) \|^2 + (1 - \nu) \theta \delta_k^2$
  balance each other

- $T_\varepsilon = \inf \{ k \geq 0 : \| \nabla f(x_k) \| < \varepsilon \}$

**Thm:** (P-Scheinberg ’18) If $pgpf > 1/2$ and $pf$ sufficiently large, then

$$E[\Phi_{k+1} - \Phi_k | \text{past}] \leq - \left( \alpha_k \| \nabla f(x_k) \|^2 + \theta \delta_k^2 \right)$$

and consequently,

$$E[T_\varepsilon] \leq O \left( \frac{1}{\varepsilon^2} \right).$$

**Proof sketch**

- $\alpha_k$ behave like random walk with $p = pgpf$
- $\{ \Phi_k \}$ supermartingales with stopping time $T_\varepsilon$
- Optional stopping time and Wald’s identity
Convex case

Assumptions:
- $f$ is convex and $\|\nabla f(x)\| \leq L_f$ for all $x \in \Omega$
- $\|x - x^*\| \leq D$ for all $x \in \Omega$

Key observation:

$$\Phi_k = \frac{1}{\nu \varepsilon} - \frac{1}{\Psi_k}$$

where $\Psi_k = \nu (f(x_k) - \inf f) + (1 - \nu) \alpha_k \|\nabla f(x_k)\|^2 + (1 - \nu) \theta \delta_k^2$

Stopping time: $T_\varepsilon = \inf\{k : f(x_k) - \inf f < \varepsilon\}$

Convergence rate, convex (P-Scheinberg ’18)

If $p_g p_f > 1/2$ and $p_f$ sufficiently large, then

$$E[T_\varepsilon] \leq O \left(\frac{1}{\varepsilon}\right)$$
$\mu$-strongly convex case

**Key observation:**

$$
\Phi_k = \log(\Psi_k) - \log(\nu \varepsilon)
$$

where

$$
\Psi_k = \nu (f(x_k) - \inf f) + (1 - \nu) \alpha_k \| \nabla f(x_k) \|^2 + (1 - \nu) \theta \delta_k^2
$$

**Convergence rate, strongly convex (P-Scheinberg '18)**

If $p_g p_f > 1/2$ and $p_f$ sufficiently large, then

$$
E[T_\varepsilon] \leq \mathcal{O} \left( \frac{L}{\mu} \log \left( \frac{1}{\varepsilon} \right) \right)
$$
Preliminary results

\[
\min_{\theta} \frac{1}{m} \sum_{i=1}^{m} \log(1 + \exp(-y_i(\theta^T x_i))) + \frac{\lambda}{2} \|\theta\|^2
\]
Open questions and extensions

Conclusions

- **General framework** for convergence results
- **Convergence analysis** (nonconvex, convex, and strongly convex) for a line search algorithm with gradient descent.
Open questions and extensions

Conclusions

- General framework for convergence results
- Convergence analysis (nonconvex, convex, and strongly convex) for a line search algorithm with gradient descent.

Applications of the stochastic process

- Line search, trust region methods (Blanchet, Cartis, Menickelly, Scheinberg ’17), and cubic regularization?
- Extensions into 2nd order stochastic methods with Hessian guarantees?

Open problems

- Finding a good practical stochastic line search for machine learning; sampling procedure too conservative
- Extending line search procedure to stochastic Wolfe conditions (BFGS)
References


